

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Žane Ban

**Spletna aplikacija za optimalen izbor inštruktorjev prve  
pomoči**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE  
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

Ljubljana, 2016



UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Žane Ban

**Spletna aplikacija za optimalen izbor inštruktorjev prve  
pomoči**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE  
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: viš. pred. dr. Aljaž Zrnec

Ljubljana, 2016



To delo je ponujeno pod licenco *Creative Commons Priznanje avtorstva - Deljenje pod enakimi pogoji 2.5 Slovenija* (ali novejšo različico). To pomeni, da se tako besedilo, slike, grafi in druge sestavine dela kot tudi rezultati diplomskega dela lahko prosto distribuirajo, reproducirajo, uporabljajo, priobčujejo javnosti in predelujejo, pod pogojem, da se jasno in vidno navede avtorja in naslov tega dela in da se v primeru spremembe, preoblikovanja ali uporabe tega dela v svojem delu, lahko distribuirajo predelava le pod licenco, ki je enaka tej. Podrobnosti licence so dostopne na spletni strani [creativecommons.si](http://creativecommons.si) ali na Inštitutu za intelektualno lastnino, Streliška 1, 1000 Ljubljana.



Izvorna koda diplomskega dela, njeni rezultati in v ta namen razvita programska oprema je ponujena pod licenco *GNU General Public License*, različica 3 (ali novejša). To pomeni, da se lahko prosto distribuirajo in/ali predelujejo pod njenimi pogoji. Podrobnosti licence so dostopne na spletni strani <http://www.gnu.org/licenses>.



Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Študenti medicine, ki sodelujejo pri projektu Za življenje, opravljajo tečaje prve pomoči v različnih krajih po Sloveniji. Ker je inštruktorjev veliko, imajo različne izkušnje in prihajajo iz različnih krajev Slovenije, se je porodila ideja o izdelavi aplikacije, ki bi omogočala pregleden in najbolj optimalen izbor inštruktorjev za posamezni tečaj glede na podane kriterije. Implementirajte spletno aplikacijo za izbor inštruktorjev za potrebe tečajev prve pomoči. Pri tem predstavite uporabljene tehnologije za izdelavo aplikacije ter zasnujte odločitveni model za izbor inštruktorjev.





## IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Žane Ban, sem avtor diplomskega dela z naslovom:

*Spletna aplikacija z odločitvenim sistemom*

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom viš. pred. dr. Aljaža Zrneca,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 14. marec 2016

Podpis avtorja:



*Zahvaljujem se mentorju na fakulteti za računalništvo in informatiko v Ljubljani viš. pred. dr. Aljažu Zrnecu za vodenje, svetovanje in mentorsko podporo ter za usmeritve pri diplomskem delu in nudenju pomoči.*

*Zahvaljujem se tudi družini, ki me je podpirala in spodbujevala skozi celoten dodiplomski študij in mi omogočila, da sem prišel do zaključnega pisanja diplomskega dela.*

*Zahvala gre tudi prijateljem, ki so me bodrili in motivirali, ko je bilo to najbolj potrebno ter projektu Za življenje, ki mi je omogočil sodelovanje in razvoj diplomskega dela.*

# Kazalo

## Povzetek

## Abstract

<b>Poglavje 1</b>	<b>Uvod.....</b>	<b>19</b>
1.1	Opis problema .....	19
<b>Poglavje 2</b>	<b>Spletna aplikacija.....</b>	<b>21</b>
2.1	Funkcionalnosti aplikacije.....	21
2.2	Predstavitev zaslonskih mask aplikacije .....	22
<b>Poglavje 3</b>	<b>Razvoj spletne aplikacije .....</b>	<b>34</b>
3.1	Uporabljena okolja za razvoj programske opreme.....	34
3.1.1	Sublime Text 2 .....	34
3.1.2	LAMP.....	34
3.1.3	XAMPP.....	34
3.1.4	phpMyAdmin .....	35
3.2	Uporabljene tehnologije .....	35
3.2.1	HTML .....	35
3.2.2	PHP .....	35
3.2.3	JavaScript.....	36
3.2.4	jQuery.....	36
3.2.5	Google Maps JavaScript API.....	37
3.2.6	CSS.....	38
3.2.7	MySQL.....	39
<b>Poglavje 4</b>	<b>Odločitveni sistem .....</b>	<b>40</b>
4.1	Varianta .....	40

4.2	Ocenitev kriterijev .....	42
4.3	Funkcija koristnosti.....	43
4.4	Rangiranje in izbira variant.....	44
<b>Poglavje 5 Sklepne ugotovitve.....</b>		<b>45</b>



## Seznam uporabljenih kratic

kratica	Angleško	slovensko
<b>HTML</b>	Hyper Text Markup Language	jezik za označevanje nadbesedila
<b>PHP</b>	Hypertext Preprocessor	tričrkovni rekurzivni akronim
<b>MySQL</b>	My Structured Query Language	metoda podpornih vektorjev
<b>CSS</b>	Cascading Style Sheets	kaskadne stilske podobe
<b>LAMP</b>	Linux, Apache, MySQL, PHP	Linux, Apache, MySQL, PHP
<b>AED</b>	Automated external defibrillator	Avtomatski eksterni defibrilator
<b>TPO</b>	Cardiopulmonary resuscitation	Temeljni postopki oživljanja





## **Povzetek**

Namen diplomske naloge je olajšati izbiro želenih inštruktorjev za prihajajoče tečaje študentov medicine s pomočjo večkriterijskega odločitvenega sistema. Cilj diplomskega dela je izdelava spletne aplikacije z implementacijo večparameterskega odločitvenega sistema, ki bi na podlagi različnih kriterijev olajšal izbor najboljših variant. Spletna aplikacija omogoča uporabniku pregledovanje prihajajočih in preteklih tečajev ter vpogled v število že opravljenih tečajev. Aplikacija uporabnikom omogoča tudi dodajanje novih tečajev, katerim se na podlagi registriranih inštruktorjev dodeli najprimernejše inštruktorje za izvedbo tečaja, o čemer slednje tudi obvesti. Aplikacija je mišljena kot pripomoček pri odločanju o tem, kdo bo sestavljal ekipo inštruktorjev v posameznem tečaju.

**Ključne besede:** večparametrski odločitveni sistem, kriterij, spletna aplikacija

## **Abstract**

Purpose of this thesis is to facilitate the selection of desired instructors for upcoming courses of medical students through the multi-criteria decision-making system. The aim is to develop web application with implementing a multi-parameter decision support system on the basis of certain criteria to facilitate selection. The web application allows the user to review upcoming and past courses gives him insight into the number of courses already carried out. The application also enables users to add new courses, to which the most suitable registered instructors are assigned to and they also receive notification of their assigned courses. The application is meant as an accessory in deciding who will be in a team of instructors in a particular course.

**Keywords:** decision-making system, criterion, web application

# **Poglavje 1     Uvod**

## **1.1    Opis problema**

Danes je ročno vnašanje podatkov zamudno in nepraktično, še posebej če je podatkov veliko in se mnogokrat ponavljajo. Prav tako je z izbiro najbolj primernih in optimalnih rešitev. Človek si je vedno znal pomagati z različnimi načini pri težkih odločitvah, da je prišel do najboljšega rezultata. Ker pa nam danes tehnologije omogočajo lažje delo, nam je uporaba računalnikov korenito spremenila tudi področja odločanja.

Čas, ki ga porabimo za pregledovanje podatkov, primerjanje vseh možnosti med saboj in nato še izbira najboljše, je bistveno daljši, če vse to opravljamo ročno, kot pa če se poslužujemo odločitvenih sistemov. Z današnjo tehnologijo je to omogočeno, ker nam olajša delo in prihrani čas, saj imamo ponekod toliko podatkov, da je že sam pregled le teh praktično nemogoč.

Današnja tehnologija nam pomaga pri odločitvah, kjer je potrebna točnost in optimalnost. Odločitveni sistemi povečujejo učinkovitost, produktivnost in uspešnost dobrega odločanja, saj vodijo k optimalni izbiri tvariant. Vendar pa se moramo zavedati, da odločitveni sistemi ne nadomeščajo človeške izbire, ampak služijo kot pomagalo pri končni odločitvi.

Odločitveni sistem je računalniški sistem, ki ponavadi vključuje matematične modele, uporabniški vmesnik in informacijske podatkovne baze, da bi zagotovil najboljše odločitve. Odločitveni sistem je podoben poslovnemu informacijskemu sistemu saj uporabniku zagotavlja informacije, poizvedovanja po podatkovnih bazah in poročila vendar pa nudi tudi model za odločanje s katerim si uporabnik lahko pomaga priti do odločitev. Bistvo odločitvenega sistema je računalniški sistem, ki pomaga upraviteljem sprejemati dobre odločitve [7]. Za izdelavo odločitvenega sistema pa je potrebno dobro pretehtati kriterije, saj so ključnega pomena za dobro odločitev.

Uporabo odločitvenega sistema lahko danes izkusimo na različnih spletnih straneh, kot so IMDB, YouTube, ki priporočajo ogled filmov in video posnetkov, glede na zgodovino ogledov, lokacijo in uporabnikovih znancev.

Glavna osrednja naloga mojega diplomskega dela je bila tako izdelava spletne aplikacije z odločitvenim sistemom, ki bi podpirala aktivnosti v okviru projekta »Za življenje«, študentov medicine v Ljubljani. Projekt Za življenje organizira tečaje prve pomoči po Sloveniji, kjer študenti medicine učijo udeležence kako reagirati, ko naletijo na človeka, ki potrebuje nujno medicinsko pomoč. Inštruktorji z vsakim tečajem, kjer predavajo, pridobijo točke, katere jim kasneje pomagajo priti do nekaterih medicinskih študentskih izmenjav in specializacij.

Do sedaj so termine in inštruktorje na posameznih tečajih planirali s pomočjo Excel tabele, kamor so se podatki vnašali ročno. Administrator je na podlagi pregleda tabele izbral ustrezne inštruktorje in jih o tečaju tudi obvestil. Sedanji sistem s tabelo Excel je zamuden, nepregleden in nepraktičen, saj saj so imeli vsi inštruktorji dostop do datoteke in pravice do spreminjanja. Zgodilo se je, da so kakšnega inštruktorja izbrisali iz tečaja, ali pa se je kakšen inštruktor, ki ni predaval na tečaju nepravilno dodal na seznam z goljufivim namenom pridobivanja točk. Poleg tega pa je bilo potrebno vsem inštruktorjem ročno vpisovati tečaje, pri katerih so sodelovali. Za vsak tečaj se je moral administrator odločiti, katere inštruktorje bo izbral za določen tečaj. Za inštruktorje uporaba Excel tabele ni bila najbolj primerna, zato je bil izziv izdelati spletno aplikacijo, ki bi nadomestila obstoječi sistem. Tečaji, ki se v okviru projekta organizirajo, so razdeljeni v štiri skupine, katere se razlikujejo med seboj po vsebini, ki so naslednje

- upravljanje z defibrilatorjem ali AED,
- temeljni postopki oživljanja ali TPO,
- ustavljanje krvavitev ter
- različni položaji nujne medicinske pomoči.

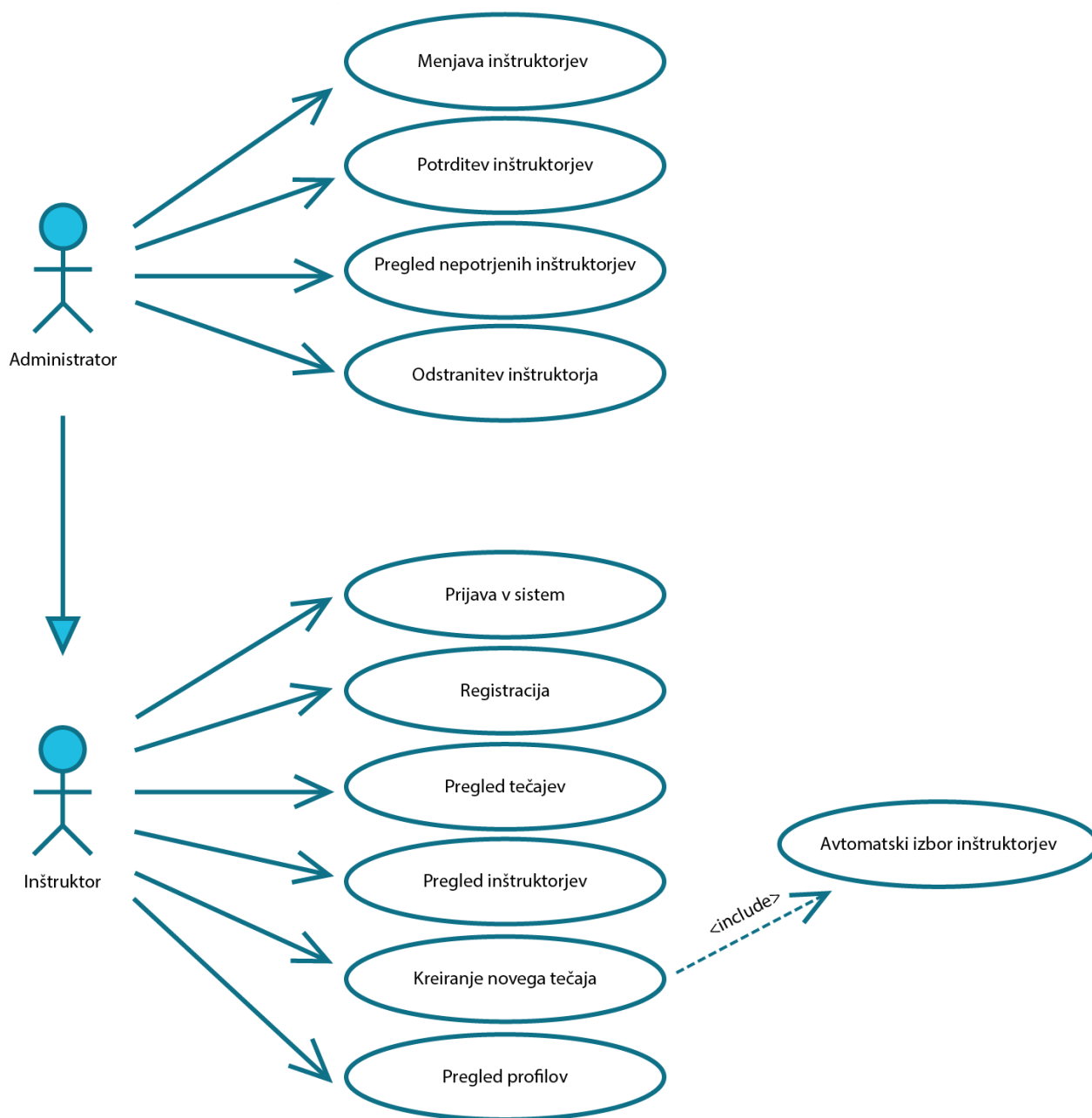
Tečaj ima pod nadzorom vodja ali koordinator tečaja, ki izbere inštruktorje, organizira prevoz inštruktorjev do kraja izvajanja tečaja in kontaktira podjetje oz. ustanovo, kjer bo tečaj izveden. V vsaki skupini je lahko en ali več inštruktorjev, odvisno od zahtevnosti posameznega tečaja in velikosti sodelujoče skupine.

Spletna aplikacija omogoča registracijo za nove inštruktorje, pregled tečajev, dodajanje novih tečajev, pregled vseh inštruktorjev in njihovih profilov, omogoča dodajanje odsotnosti, avtomatsko izbiro inštruktorjev ter možnost ročne spremembe izbranih inštruktorjev, ki so bili izbrani z uporabo vgrajelega odločitvenega sistema. Vsebovani odločitveni za avtomatski izbor inštruktorjev in preostale funkcionalnosti bodo podrobno predstavljene v nadaljevanju diplomskega dela.

## **Poglavje 2     Spletna aplikacija**

### **2.1     Funkcionalnosti aplikacije**

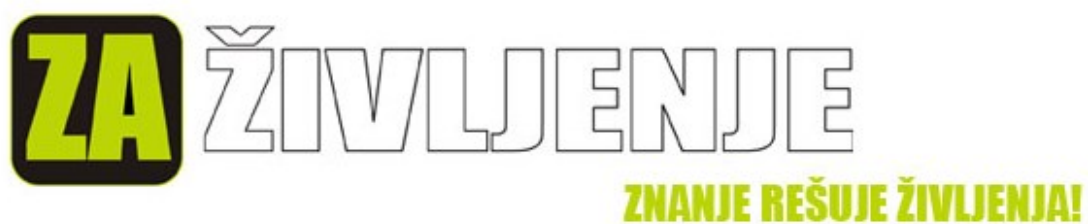
Funkcionalne zahteve aplikacije prikazuje diagram primerov uporabe na Sliki 1. Iz diagrama lahko vidimo, da funkcionalnosti aplikacije uporabljata dva akterja. Inštruktor, ki se lahko registrira, prijavi v sistem, pregleda tečaje in inštruktorje ter doda nove tečaje. Administrator pa pa lahko poleg funkcionalnosti, ki jih uporablja inštruktor, uporablja še naslednje funkcionalnosti: zamenja inštruktorje, ki jih je za nek tečaj izbral odločitveni sistem ter sprejme ali zavrne novo registrirane člane. Vse na novo registrirane inštruktorje administrator ročno pregleda, s čimer se zagotovi, da v sistemu ni lažnih inštruktorjev. Glavna funkcionalnost spletne aplikacije pa je avtomatski izbor inštruktorjev, ki temelji na uporabi odločitvenega modela, katerega vam bomo podrobneje predstavili v poglavju 4.



Slika 1. Diagram primerov uporabe.

## 2.2 Predstavitev zaslonskih mask aplikacije

Ko inštruktor odpre zahtevani spletni naslov aplikacije, se mu prikaže začetna zaslonska maska Vstopna stran, kamor se vpiše s svojim naslovom elektronske pošte in geslom. Če svojega računa še nima, se mora najprej registrirati. To je razvidno iz Slike 2.



email
geslo

vstopi

registracija

Slika 2: Zaslonska maska - Vstopna stran.

Na zaslonski maski Registracija mora uporabnik poleg osebnih podatkov in prebivališča navesti še raven znanja, ki ga ima za neko skupino praktičnega usposabljanja, podatke o lastnem prevozu ter seveda geslo, kot je to razvidno iz Slike 3. Aplikacija tu preveri elektronski naslov, če je morda uporabnik že dodan v podatkovno bazo.

ime  
  
 priimek  
  
 email  
  
 telefonska  
  
 TRR  
  
 naslov  
  
 poštna številka in kraj  
  
 velikost majice  
 ▾  
☐ Imam lasten prevoz  
 spol  
 ▾  
☐ Vodja  
 znanje AED  
 ▾  
 znanje TPO  
 ▾  
 znanje krvavitve  
 ▾  
 znanje položaj  
 ▾  
 datum rojstva  
  
 geslo  
  
 ponovno vpišite geslo

Slika 3. Zaslonska maska – Registracija.

Ko se inštruktor uspešno registrira, potrebuje za vstop v spletno aplikacijo potrditev s strani administratorjev. Če je njegov profil še nepotrjen, ga aplikacija ob prijavi preusmeri nazaj na prvo stran, kar preprečuje prijavo v sistem nepridipravnim uporabnikom.




Na vseh zaslonkih maskah lahko inštruktor vidi glavo zaslonske maske, kjer ima možnost izbire različnih zaslonskih mask kot so zaslonska maska Tečaji, zaslonska maska Pretekli tečaji, zaslonska maska Moj profil, zaslonsko masko Top 10, ter zaslonska maska Člani. V glavi zaslonske maske lahko vidi tudi število tečajev pri katerih je že sodeloval, kar je razvidno iz Slike 4.



Slika 4: Glava zaslonske maske.

Ko pa inštruktorja administrator potrdi in doda v sistem, inštruktor vidi po prijavi zaslonsko masko Tečaji. Tu ima pregled nad prihajajočimi tečaji, ki se še niso zaključili. Zaslonska maska tečaji, je vidna na Sliki 5, koda v nadaljevanju pa prikazuje prihajajoče tečaje, kjer je datum tačaja večji ali enak trenutnemu datumu.

```
$check="SELECT * FROM tecaji WHERE datum >= CURDATE() ORDER BY datum ";
mysql_query("SET NAMES 'utf8'");
$result=mysql_query($check);
while ($row = mysql_fetch_assoc($result)) {
    $sid = $row["idTecaj"];
    echo '<a href="dogodek.php?dogodek='.$sid.'">
        <div id="posamezenTecaj">
            <div id="tecajLevo">
                '.$sime.'
            </div>
            <div id="tecajSredi">
                '.$datum.'
            </div>
        </div>
    </a>';
```



# ŽIVLJENJE

ZNAJJE REŠUJE ŽIVLJENJA!

Živjo brane!

Opravljenih tečajev: 11

Tečaji
Pretekli tečaji
Moj profil
Top 10
člani


**Tečaji**

marobor 3	29.09.2015
not gonna end well	30.09.2015
Krka d.d., Novo Mesto	01.10.2015
Koper Knjiznica	16.10.2015
Drenov Grič	09.01.2016

Dodaj tečaj

Slika 5: Zaslonska maska - Tečaji.

Na spodnji strani zaslonske maske Tečaji ima inštruktor tudi možnost dodajanja novega tečaja. Tu mora vpisati ime tečaja, število zahtevanih inštruktorjev za skupine praktičnih znanj, naslov, kjer se bo tečaj izvajal ter dodaten opis, ki je namenjen podrobnostim tečaja. Tukaj lahko tudi izbere kakšno izkušnost zahteva od inštruktorjev, tj. minimalno število že opravljenih tečajev ter določi uteži, po katerih bo odločilni sistem določal ocene, kot nam to prikazuje Slika 6. Prioritete odločevalec določi od 1 do 4, kjer je 1 najpomembnejši faktor in 4 najmanj. Parametri alternative katerim inštruktor določi uteži se delijo na znanje, izkušnost, oddaljenost in lasten prevoz.



**ZA ŽIVLJENJE**  
ZNAJJE REŠUJE ŽIVLJENJA!

Živjo brane!

Opravljenih tečajev: 11

Tečaji
Pretekli tečaji
Moj profil
Top 10
člani

**Tečaj:**

**datum:**

**potrebno število koordinatorjev:**

**potrebno število inštruktorjev AED:**

**potrebno število inštruktorjev TPO:**

**potrebno število inštruktorjev krvavitve:**

**potrebno število inštruktorjev položaj:**

**Minimalno število opravljenih tečajev:**

**opis:**

**Naslov:**

Razvrsti od 1-4 po pomembnost (1 - najpomembnejši faktor)

**Znanje**

**Izkušnje**

**Oddaljenost**

**Lasten avtomobil**

Slika 6: Zaslonska maska - Dodaj tečaj.

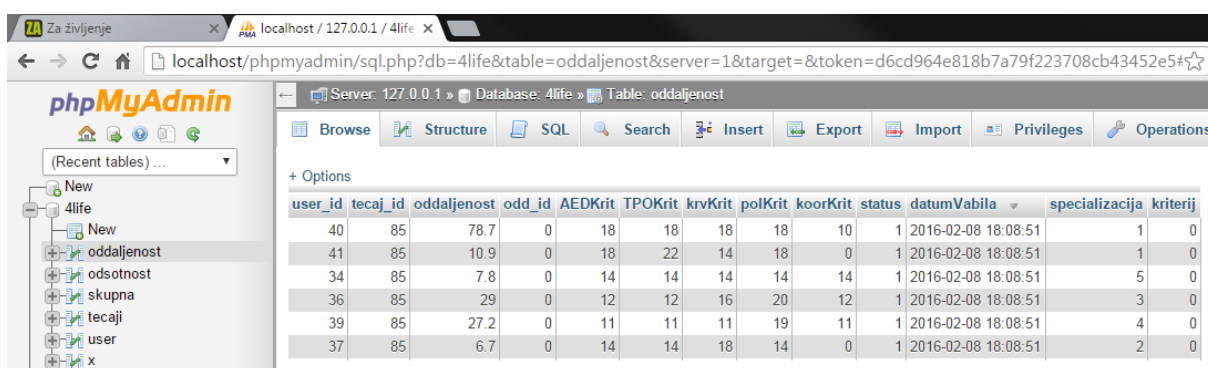
Ko izpolni vsa obvezna okenca, klikne na gumb dodaj tečaj. Sistem nato prebere vse vpisane podatke in zahteve. Spletna aplikacija pregleda bazo z inštruktorji in preveri če morda pogoji niso prezahtevni. Če so pogoji sprejemljivi, v tabelo s tečaji doda tečaj, v drugem primeru pa mora inštruktor znižati zahtevnost odločilnega sistema.

Nato sistem vsakemu od inštruktorjev izračuna oddaljenost. Za to skrbi Google Maps JavaScript API s storitvijo Distance Matrix. Vsakemu od inštruktorjev vzame domači naslov ter naslov tečaja z uporabo skriptnega jezika PHP iz baze podatkov, ter ga s pomočjo funkcije

json\_encode() poda objektu skriptnemu jeziku JavaScript. V storitvi Distance Matrix poda funkciji initMap() začetni naslov ter končni cilj skupaj z atributi, ki določajo katero pot naj funkcija izbere. Kakor je razvidno iz kode, smo kot način prevoza uporabili vožnjo z avtomobilom, podatke nam poda v metričnem sistemu, avtocestam pa se ni potrebno izogniti. Kar funkcija vrne, se zapiše inštruktorju v atribut oddaljenost v podatkovno bazo. Koda v nadaljevanju prikazuje uporabo storitve Distance Matrix, ki sprejme kot attribute začetne in končne destinacije, ter izračuna končno dolžino poti.

```
var start = <?=json_encode($start)?>;
var end = <?=json_encode($end)?>;
var service = new google.maps.DistanceMatrixService;
service.getDistanceMatrix({
    origins: [start],
    destinations: [end],
    travelMode: google.maps.TravelMode.DRIVING,
    unitSystem: google.maps.UnitSystem.METRIC,
    avoidHighways: false,
    avoidTolls: false
})
```

Ko se izračuna vse razdalje, ter jih doda inštruktorjem v podatkovno bazo, algoritem vsakemu izračuna funkcijo koristnosti za vsako skupino tečaja. Na podlagi izračunane ocene razvrsti inštruktorje od najvišje uvrščenih do najnižjih, nato preveri, da inštruktorji še niso dobili povabila za kakšno drugo skupino in jim pošlje vabilo, kar je videno na Sliki 7.

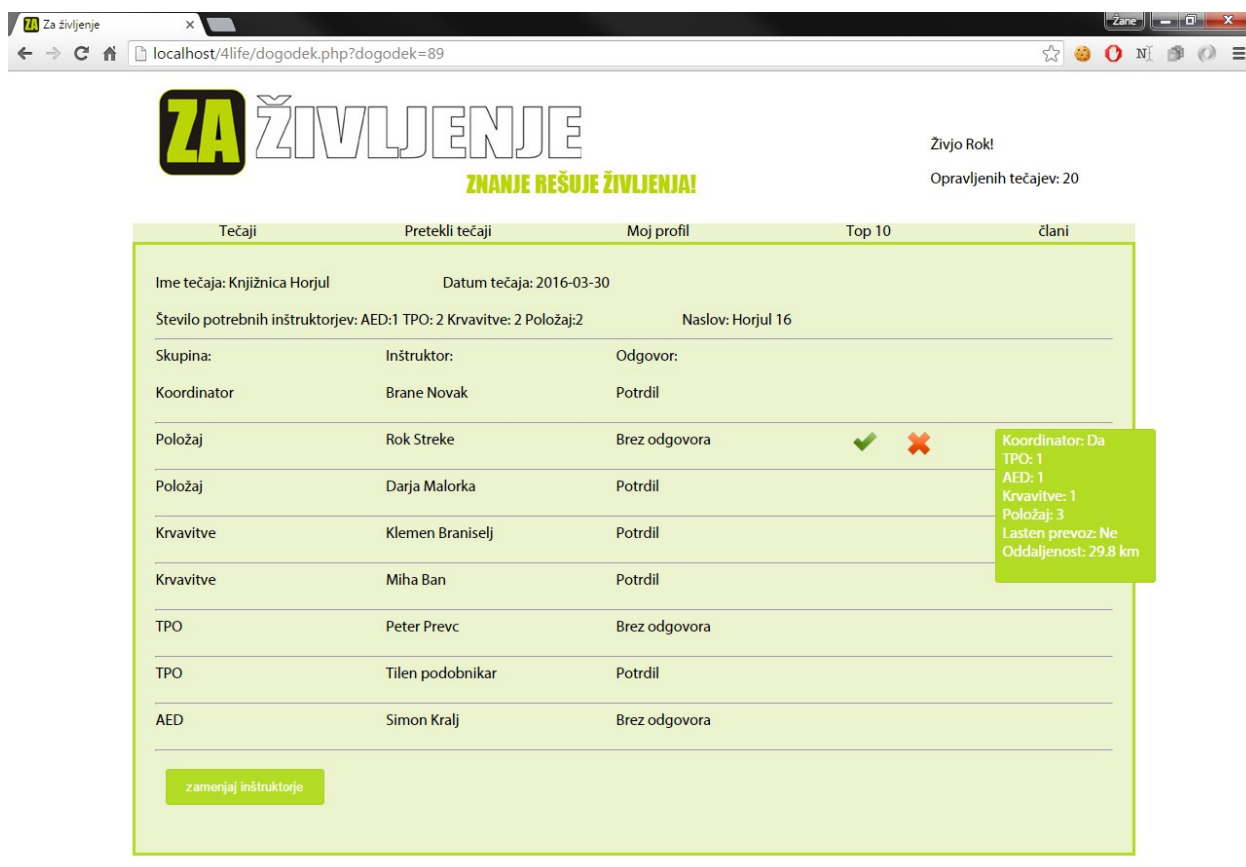


user_id	tecaj_id	oddaljenost	odd_id	AEDKrit	TPOKrit	krvKrit	polKrit	koorKrit	status	datumVabila	specializacija	kriterij
40	85	78.7	0	18	18	18	18	10	1	2016-02-08 18:08:51	1	0
41	85	10.9	0	18	22	14	18	0	1	2016-02-08 18:08:51	1	0
34	85	7.8	0	14	14	14	14	14	1	2016-02-08 18:08:51	5	0
36	85	29	0	12	12	16	20	12	1	2016-02-08 18:08:51	3	0
39	85	27.2	0	11	11	11	19	11	1	2016-02-08 18:08:51	4	0
37	85	6.7	0	14	14	18	14	0	1	2016-02-08 18:08:51	2	0

Slika 7. Tabela z izračunanimi ocenami.

V primeru, ko je kvalificiranih inštruktorjev zadosti, se pojavi zaslonska maska Tečaj na kateri lahko inštruktor pregleda, koga je odločilni sistem določil za razpisani tečaj. Zaslonska

maska nato pokaže katero skupino je sistem dodelil inštruktorjem ter če so sami že potrdili, da se bodo udeležili tega tečaja.. Ko gre inštruktor z miško čez celice kjer so razvrščeni predavatelji, se jim pokaže okence pri vsakemu posamezniku, ki prikaže kakšne attribute ima ta inštruktor. Kogar odločitveni sistem doda na seznam inštruktorjev, se mu pri njegovem imenu prikažeta dva gumba, kot prikazuje Slika 8. To sta zelena kljukica in rdeč križ. S klikom na zeleno kljukico inštruktor potrdi, da se bo tečaja udeležil in vsi ostali lahko vidijo ta odziv pri stolpcu Odgovor.



Slika 8. Zaslonska maska - Tečaj.

V primeru, ko inštruktor že potrdi, da se bo tečaja udeležil, vendar se kasneje premisli, ima še vedno možnost rdečega križa, ki ga odjavi od tega tečaja. Njegovo mesto nato dobi naslednji v vrsti. Koda v nadaljevanju prikazuje pregled inštruktorja in njegovega odziva pri tečaju.

```
if ($user_id == $cookie) {
    // ce je dobil vabilo status = 1 se lahko prijavi ali odjavi
    if ($status == 1){
        echo '
    
```

```

        <form method="post" action="plus.php?id='.$tečaj.' ">
            <input type="image" onclick="show('comment')>
            src="up.png" name="image" align="middle" id="up">
        </form>
        ';
    }
    else{
        echo '
            <form method="post" class = "hidden"
            action="plus.php?id='.$tečaj.' ">
            <input type="image" onclick="show('comment')>
            src="up.png" name="image" align="middle" id="up">
        </form>
        ';
    }
    // ce je dobil vabilo status = 1 ali se je že prijavil status = 2 se lahko odjavi
    if ($status == 1 || $status == 2) {
        echo '
            <form method="post" action="minus.php?id='.$tečaj.' ">
            <input type="image" onclick="show('comment')>
            src="down.png" name="image" align="middle" id="down">
        </form>
        ';
    }
    else{
        echo '
            <form method="post" class="hidden"
            action="plus.php?id='.$tečaj.' ">
            <input type="image" onclick="show('comment')>
            src="up.png" name="image" align="middle" id="up">
        </form>
        ';
    }
}

```

Če pride do primera, da se z odločilnim sistemom administrator ne strinja, in želi nekatere inštruktorje zamenjati, se lahko sam odloči, koga bo zadolžil skupinam. Na spodnji strani zaslonske maske Tečaj je lociran gumb Zamenjaj inštruktorje, ki odpre zaslonsko masko Ročno upravljanje z inštruktorji. Na tej zaslonski maski administrator vidi stopnjo znanja, ki ga inštruktorji premorejo, število tečajev na katerih so že sodelovali in oddaljenost od tečaja podano v kilometrih, kot prikazuje Slika 9.

Tečaji		Pretekli tečaji				Moj profil				Top 10		člani	
Ime tečaja: Krka d.d., Novo Mesto						Datum tečaja: 2015-10-01							
Število potrebnih inštruktorjev: AED:2 TPO: 2 Krvavitve: 0 Položaj:1						Naslov: Novo Mesto							
Inštruktor:		Znanja:				Izkušnost				Oddaljenost			
Ime in priimek:	Vodja:	AED:	TPO:	Krvavitve:	Položaj:	Vodja:	AED:	TPO:	Krvavitve:	Položaj:	KM:		
Žane Ban	-	srednja	-	nižja	višja	-	1	-	0	0	0	Izberi	ok
ana banana	-	-	-	nižja	srednja	-	-	-	0	2	171	Izberi	ok
brane banane	-	višja	srednja	-	-	-	1	0	-	-	118	Izberi	ok
Tim Barkos	nižja	-	-	višja	srednja	0	-	-	1	0	0	Izberi	ok
brane branko	nižja	srednja	-	-	-	2	0	-	-	-	0	Izberi	ok
Peter Bratko	nižja	-	srednja	-	-	0	-	5	-	-	118	Izberi	ok
barbara fasa	nižja	-	-	-	srednja	1	-	-	-	3	71	Izberi	ok
france francoz	-	nižja	nižja	-	višja	-	2	0	-	0	1	Izberi	ok
jože kovač	-	nižja	nižja	-	-	-	3	0	-	-	97.2	Izberi	ok
Lilo Lilovski	nižja	-	-	nižja	višja	2	-	-	5	0	80.3	Izberi	ok
Miha Miha	nižja	-	-	-	nižja	0	-	-	-	1	833	Izberi	ok
Žiga Most	-	-	nižja	-	-	-	-	1	-	-	77.3	Izberi	ok
Klemen Novak	nižja	nižja	nižja	nižja	nižja	3	0	0	0	0	71	Izberi	ok
Janez Novak	-	-	-	nižja	-	-	-	-	1	-	79.7	Izberi	ok
Katarina Remec	nižja	nižja	-	srednja	-	2	0	-	3	-	80.2	Izberi	ok
Žiga Žaga	-	srednja	-	-	nižja	-	0	-	-	0	64.6	Izberi	ok

Slika 9. Zaslonska maska - Ročno upravljanje z inštruktorji.

Na zaslonski maski Pretekli tečaji ima inštruktor razvrščene tečaje, ki so se že zgodili, za lažje vodenje evidence, saj so tu zapisani tudi inštruktorji, ki so predavali. Vsak član ima tudi svojo profilno zaslonsko masko, kjer so izpisani vsi podatki, ki jih je vnesel inštruktor ob registraciji. Ima pa tudi možnost dodajanja odsotnosti. Ko klikne v okenca za odsotnost, se mu prikažeta dva koledarja, na katerih izbere, od kdaj do kdaj bo odsoten in ga kasneje odločitveni sistem ne upošteva, če pride kakšen tečaj na vmesni datum. Če ima administrativne pravice, pa ima možnost nekega člana izločiti iz spletne aplikacije, če inštruktor ni več aktiven, oziroma ni to za kar se izdaja. Primer obojega lahko vidimo na Sliki 10.


**ŽIVLJENJE**  
 ZNANJE REŠUJE ŽIVLJENJA!

Živjo branel  
 Opravljenih tečajev: 11

Tečaji    Pretekli tečaji    Moj profil    Top 10    člani

Klemen Novak  
 email: kn0123@student.uni-lj.si  
 Število opravljenih tečajev: 10  
 Telefonska številka: 040 123 456  
 Odsotnost  
 od   
 do   
 Dodaj odsotnost

Datum rojstva: 01.09.1992  
 Transakcijski račun: 0214 1264 5789 654  
 Naslov: Ljubljana 2  
 Spol: Moški  
 Velikost majice: M

odstrani uporabnika  


Slika 10. Zaslonska maska - Profil.

Zaslonska maska Top 10 je narejena zgolj za stimulacijo študentov, saj je na njem lestvica desetih najbolj zagnanih inštruktorjev, ki so sodelovali pri največ tečajih, kot je razvidno iz Slike 11.


**ŽIVLJENJE**  
 ZNANJE REŠUJE ŽIVLJENJA!

Živjo Miha!  
 Opravljenih tečajev: 4

Tečaji    Pretekli tečaji    Moj profil    Top 10    člani

1. Rok Streke 20
2. Peter Klemen 11
3. Blaž Oblak 10
4. Peter Prevč 9
5. Tilen Podobnikar 8
6. Brane Novak 7
7. Simon Kralj 6
8. Miha Ban 5
9. Klemen Braniselj 2
10. Darja Malorka 1

Slika 11. Zaslonska maska – Top 10.



Ostane le še zaslonska maska Člani, kjer so razvrščeni vsi inštruktorji. S klikom na njihovo ime inštruktor pride do njihovih profilov, kjer si lahko ogleda podatke posameznih inštruktorjev. Na spodnji strani zaslonske maske Člani je gumb, ki ga vidijo le inštruktorji z administrativnimi pravicami in omogoča dodajanje novih članov v sistem, ki so se registrirali.

## **Poglavje 3      Razvoj spletne aplikacije**

Spletno aplikacijo smo razvili z uporabo paketa XAMPP za postavitev PHP spletnega strežnika Apache in MySQL strežnika. Vse skupaj je bilo za potrebe razvoja in testiranja nemeščeno na lokalnem strežniku.

### **3.1      Uporabljena okolja za razvoj programske opreme**

#### **3.1.1 Sublime Text 2**

Sublime Text 2 je več platformni urejevalnik izvirne kode. Podpira več različnih programskih in označevalskih jezikov, kateri ponuja tudi razširitve s pomočjo različnih vtičnikov [10]. V okviru izdelave aplikacije je bil uporabljen za pisanje izvirne kode HTML, PHP, JavaScript, CSS in SQL.

#### **3.1.2 LAMP**

Med pakete LAMP spadajo odprtokodne programske opreme, s katerimi lahko postavimo strežnik, kateri je sposoben gostiti dinamične spletne strani. Kratica LAMP predstavlja operacijski sistem Linux, spletni strežnik Apache, MySQL strežnik podatkovne baze MySQL in skriptni jezik PHP. Obstajajo tudi različice, kot so WAMP (Microsoft Windows), MAMP (Macintosh) ali SAMP (Solaris) [11].

#### **3.1.3 XAMPP**

XAMPP nam omogoča zelo hitro namestitev spletnega strežnika saj spada med pakete LAMP, kar pa pove že samo ime. X predstavlja križ, kot cross-platform, kar pomeni da deluje na vseh platformah. A pomeni Apache, M predstavlja MySQL, prvi P predstavlja PHP, zadnji P pa Perl[12]. Omogoča postavitev modulov kot so Apache, MySQL, FileZilla, Mercury in Tomcat. V diplomski nalogi smo ga uporabili za postavitev spletnega strežnika Apache in spletnega odjemalca za podatkovno bazo MySQL.

##### **3.1.3.1      Spletni strežnik Apache HTTP**

Apache je svetovno najbolj razširjen strežnik HTTP. Apache je ključen pri širjenju spleta, saj je od druge polovice leta 2013 na njem postavljenih več kot 54% vseh spletnih strani. Ime so izbrali na podlagi spoštovanja do plemena ameriških indijancev Apache in njihovih

superiornih veščin vojskovanja in strategij. Pri razvijanju spletne aplikacije smo ga uporabili za postavitev lokalnega strežnika.

### **3.1.4 *phpMyAdmin***

Brezplačno programsko orodje phpMyAdmin je napisano v jeziku PHP, ki smo ga uporabili za ravnanje z MySQL administracijo preko spleta. Podpira širok spekter operacij pri MySQL in MariaDB. Pogosto uporabljeni postopki kot so upravljanje s podatkovnimi bazami, tabelami, stolpci, relacijami, indeksi, uporabniki ali dovoljenji, se lahko izvedejo preko uporabniškega vmesnika, medtem ko imamo še vedno možnost, da neposredno izvršimo katerikoli stavek SQL [13].

## **3.2 Uporabljene tehnologije**

V nadaljevanju so predstavljene tehnologije, ki so bile uporabljene pri izdelavi spletne aplikacije.

### **3.2.1 *HTML***

Jezik za označevanje nadbesedila, je označevalni jezik, katerega smo uporabili za izdelavo jedra spletne strani. Predstavlja osnovo spletne strani, saj se z njim tvori ogrodje HTML. Je označevalni jezik in je sestavljen iz značk, ki so zapisane v špičastih oklepajih. Značke predstavljajo HTML elemente, ki so običajno v parih. Znotraj značk imamo lahko tudi gnezdenje le teh.

HTML omogoča objavo besedilnih dokumentov, tabel, fotografij, audio in video posnetkov, preusmeritve na druge strani, pošiljanje elektronske pošte in še mnogo drugih možnosti. Dokument HTML osnovno radelimo z značkami kot so <head> in <body>. V tako imenovano glavo spadajo informacije dokumenta in se na strani ne izpisujejo. Vključujejo povezave na datoteke kot so CSS, JavaScript, jQuery. V tako imenovano jedro dokumenta pa spada vsebina spletne strani.

### **3.2.2 *PHP***

PHP je razširjen skriptni jezik, ki smo ga v diplomskem delu uporabili za izdelavo dinamičnega dela spletne strani. Med drugim tudi za pošiljanje elektronske pošte, kot prikazuje primer sintakse v nadaljevanju. Jezik PHP deluje na strežniku in ponuja podporo dostopa do podatkovne baze. Strežnik prejme za vhod PHP kodo ter zgenerira spletno stran, ki jo ponudi kot izhod. PHP koda se nahaja med oznakama »<?php« in »?>«. V PHP

datotekah se poleg kode PHP lahko nahajajo tudi druge kode kot so HTML, CSS, JavaScript...

Uporabljamo lahko tudi različne funkcije, ki so že definirane, kot na primer `include()`, ki na novo stran vključi že prej napisano kodo ali pa `mysqli_connect()`, s katero se povežemo na MySQL strežnik. Lahko pa deklariramo tudi nove funkcije. Spremenljivke lahko pošiljamo med različnimi stranmi s HTML elementom `<form>`, ki jih shrani v spremenljivke `$_GET` ali `$POST`. Metoda GET pošlje podatke preko URL naslova, kar je praktično kadar potrebujemo povezave do različnih strani, vendar pa je ne uporabljamo za pošiljanje prevelikega števila podatkov, še zlasti pa ne pri posredovanju gesel in drugih zaupnih podatkov. Metoda post, pa svoje podatke skriva in je tako bolj primerna za pošiljanje velikih količin podatkov, prav tako pa tudi za posredovanje zaupnih podatkov [3].

Primer sintakse:

```
$from = 'zazivljenje@gmail.com';
$subject = 'Za življenje ti pošilja novo sporočilo. ';
$message = "Sporočilo, katere vrstice so krajše od 70 znakov. ";
}
// vrstice sporočila ne smejo presegati 70 znakov (pravilo PHP)
$message = wordwrap($message, 70);
$header = 'From: zazivljenje@gmail.com\r\n';
$retval = mail ('banzane@gmail.com', $subject, $message, $header);

if( $retval == true )
{
    echo "Message sent successfully...";
}
else
{
    echo "Message could not be sent...";
```

### 3.2.3 JavaScript

JavaScript je dinamični, visokonivojski, objektni skriptni jezik, ki ga je razvil Netscape. Uporabili smo ga za pomoč pri implementaciji interaktivnih delov spletne strani. JavaScript vključimo v HTML, ko želimo na strani opravljati naloge, katerih ne moremo opraviti le na statični strani [2].

### 3.2.4 jQuery

jQuery je knjižnica napisana s skriptnim jezikom JavaScript, ki vsebuje funkcije za hitrejši razvoj spletnih aplikacij. Kot pravi njihov slogan "Write Less, Do More" je bistvo te

knjižnice, da z manj kode naredimo več. Z uporabo kratkih klicev funkcij, lahko s tem ogrođjem na strani dodamo veliko animacij ter vizualnih učinkov, za katere bi drugače potrebovali veliko več časa. Rešuje pa tudi težave z brskalniki, kjer različni brskalniki prikazujejo enako stvar na drugačen način [7]. Knjižnico jQuery smo v diplomski nalogi med drugim uporabili za lažjo izbiro datumov na spletni aplikaciji, kot prikazuje sintaksa v nadaljevanju.

Primer sintakse:

```
<script type="text/javascript" src="jquery-1.2.6.min.js">
    $(function() {

        $("#datepicker").datepicker();

    });
</script>
```

### 3.2.5 Google Maps JavaScript API

Storitev Google's Distance Matrix smo uporabili za izračun oddaljenosti od naslova inštruktorja do naslova, kjer se bo izvajal tečaj, saj ta storitev izračuna potovalno razdaljo in trajanje potovanja med mnogimi začetnimi in končnimi destinacijami. Dostop do te storitve je asinhron, saj Google Maps API potrebuje klic z zunanjim strežnikom. Zaradi tega, je potrebno podati metodo s povratnim klicem (callback), da po zaključku zahteve izvede obdelavo rezultatov.

Do storitve Distance Matrix je možno dostopati preko objekta google.maps.DistanceMatrixService. Za sprožitev zahteve za storitve Distance Matrix je potrebno klicati metodo DistanceMatrixService.getDistanceMatrix(), katera poda objektu začetno in končno destinacijo ter način potovanja,

Objekt poda storitvi naslednja polja:

- *origins* - seznam enega ali večih začetnih naslovov, od koder se bo računala razdalja in potovalni čas, lahko je v obliki besede (string) ali pa google.maps.LatLng, ki poda koordinatno dolžino in širino,
- *destinations* - prav tako seznam enega ali večih naslovov do koder se bo računala razdalja in potovalni čas, lahko je v obliki besede (string) ali pa google.maps.LatLng, ki poda koordinatno dolžino in širino,
- *travelMode* - način prevoza pri računanju potovalne smeri,

- *unitSystem* - uporabimo ga, ko bomo prikazovali razdaljo, tukaj lahko izbiramo med metričnim ali imperialnim sistemom,
- *durationInTraffic* - določa ali naj elementi upoštevajo trenutne prometne razmere k trajanju potovanja, tukaj bomo dobili rezultat le, če je ta podatek na voljo v zahtevanem območju,
- *avoidHighways* - s tem določimo, če želimo da se preračunana pot izogne avtocestam med začetno in končno destinacijo,
- *avoidTolls* – s tem določimo, če želimo da se preračunana pot izogne cestninam med začetno in končno destinacijo.

### 3.2.5.1 Travel modes

Ko računamo čas in razdaljo, lahko določimo kakšen način prevoza bomo izbrali. Trenutni omogočeni načini prevoza so:

- *google.maps.TravelMode.BICYCLING* - zahteva vožnjo s kolesom po kolesarskih stezah ter po prednostnih ulicah (trenutno na voljo le v ZDA in nekaterih kanadskih mestih)
- *google.maps.TravelMode.DRIVING* - je privzeto nastavljen način prevoza in izbere standardne cestne poti s pomočjo cestnega omrežja
- *google.maps.TravelMode.TRANSIT* - zahteva smer preko javno prevoznih smeri
- *google.maps.TravelMode.WALKING* - zahteva sprehajalne smeri po pešpoteh in pločnikih, kjer so na voljo

Uspešen klic storitve Distance Matrix vrne objekta *DistanceMatrixResponse* in *DistanceMatrixStatus*. Pošlje se ju povratnemu klicu funkcije, ki smo jo določili v zahtevku.

Objekt *DistanceMatrixResponse* vsebuje razdaljo in trajanje za vsak par začetnih in končnih destinacij, pri katerih je možno izračunati pot **Error! Reference source not found.**

### 3.2.6 CSS

CSS je kratica za Cascading Style Sheets in določa, kako bo neka spletna stran na koncu izgledala v brskalniku. Če so ostale našteje tehnologije skrbele za vsebino spletne strani, ta skrbi za obliko in izgled. Skrbi za barve, odmike, pisave in še mnogo več. Poleg tega lahko

spreminjamo obliko strani na podlagi pozicije inštruktorjeve miške. Potrebno oblikovanje lahko ločimo v svoj dokument, katerega potem iz HTML dokumenta kličemo po potrebi [4]. Tako se znebimo tudi ponavljanja kode. CSS smo med drugim uporabili tudi za prikaz parametrov ko gremo z miško nad inštruktorjevo ime v zaslonski maski Tečaj, kot prikazuje primer sintakse v nadaljevanju.

Primer sintakse:

```
#hoverOblak{
    float: right;
    opacity: 0;
    width: 150px;
    height: 150px;
    -webkit-border-radius: 3;
    -moz-border-radius: 3;
    border-radius: 3px;
    color: #ffffff;
    margin: -1%;
    padding: 1% 2% 1% 2%;
    border: solid #accf2d 1px;
    text-decoration: none;
}
```

### 3.2.7 MySQL

MySQL je odprtokodna implementacija relacijske podatkovne baze, ki uporablja jezik SQL za upravljanje s podatkovnimi bazami in deluje na principu odjemalec - strežnik. Je relacijska zbirka podatkov, ki deluje kot samostojen strežnik ali pa skupaj z drugimi tehnologijami [9]. V diplomski nalogi smo jo uporabili za poizvedovanje in zapise v podatkovno bazo.

## Poglavje 4      Odločitveni sistem

Odločitveni sistem smo uporabili za optimalno izbiro inštruktorjev, ki bodo sodelovali na tečaju. Pri odločanju je potrebno izmed različnih variant (alternativ, inčic, možnosti) izbrati tisto, ki se najboljše ujema s postavljenim ciljem oziroma zahtevam. Za učinkovito sprejemanje odločitev, moramo biti sposobni predvidevati izid tako dobrih kot slabih lastnosti variant in pretehtati, katera možnost je najboljša za dani problem. Poleg izbora najboljše variante, včasih želimo inčice tudi rangirati od najboljše do najslabše [14]. Pomembna lastnost odločilnega sistema je tudi, da omogoča pojasniti kako smo prišli do dane izbire. Prav tako nas zanima v kakšni meri odločitev izpolnjuje določene cilje, ter kako je proces odločitve prispeval h končni izbiri [1].

### 4.1      Varianta

Varianta oz. možnost, alternativa, različica je izbira, ki jo imamo na voljo pri odločilnem modelu. To so objekti, ljudje, aktivnosti, ki so med seboj podobni in jih lahko primerjamo med seboj. Med izrazi praviloma ne delamo razlik, kajti vsi predstavljajo neko izbiro, ki se med seboj izključujejo [1].

Ko primerjamo variante in jih ocenjujemo, ponavadi te nimajo le ene same lastnosti. Na primer, pri nakupu prenosnega računalnika nas ne zanima le cena, temveč tudi zmogljivost, velikost zaslona, teža, itd. Takšno odločanje imenujemo večparametrsko, kajti parameter je ena izmed lastnosti, katere ima neka varianta [1].

V našem primeru je varianta človek - inštruktor, ki ima naslednje, za nas pomembne lastnosti:

1.    znanje,
2.    izkušnje,
3.    oddaljenost in
4.    lasten prevoz.

Lastnost Znanje opisuje raven znanja za neko tematiko, ki se izvaja na tečaju. Tematike so naslednje: AED, TPO, zaustavljanje krvavitev ter različni položajih pri ponesrečencev, kateri so bile omenjene že v uvodu.

Lastnost Izkušnje pove, kolikšno je število tečajev iz neke teme, pri katerij je inštruktor že sodeloval.



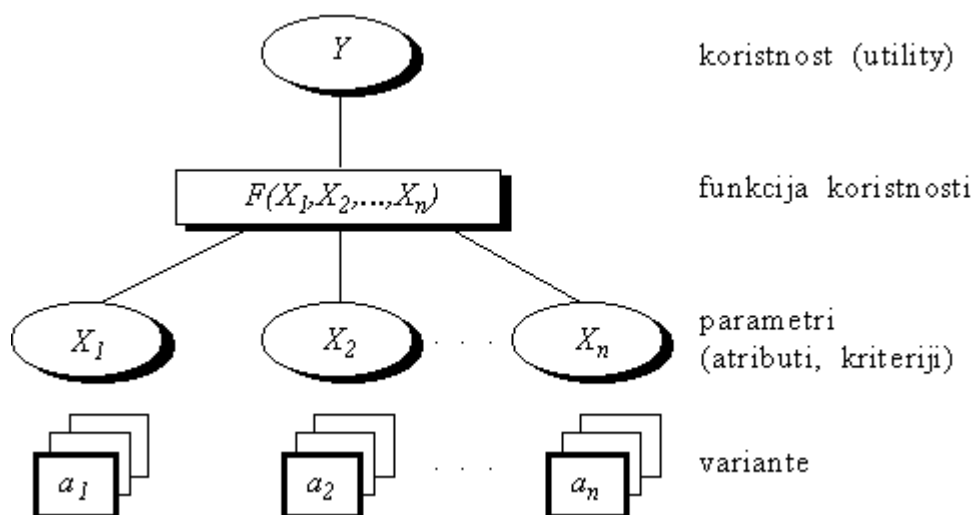
Lastnost Oddaljenost pove koliko kilometrov je inštruktorjev naslov prebivališča oddaljen od naslova, kjer se bo opravljal tečaj.

Lastnost Prevoz je zadnji parameter, ki nam pove ali ima inštruktor svoj lasten prevoz.

Ko govorimo o odločitvenih modelih, se moramo najprej odločiti katerega bomo uporabili. Med seboj se razlikujejo po zahtevnosti, pogostosti, negotovosti, strukturiranosti, številu kriterijev... Med slednje spada tudi večparametrski odločitveni model, ki ga uporabimo, ko nas zanima več kot le ena lastnost alternativ [5]. Ta model temelji na razbijanju problema na manjše podprobleme. Tukaj vzamemo variante in vzamemo vsako njihovo lastnost ter jo ločeno ocenimo. Vse skupaj na koncu združimo in dobimo oceno, ki jo lahko rangiramo, kot vidimo na Sliki 12.

Pri odločitveni analizi je pristop sistematičen, ko rešujemo odločitvene probleme, kajti odločitveni proces imamo razdeljen na [1]:

1. identifikacija odločitvenega problema
2. identifikacija alternativ
3. razgradnja problema in modeliranje
4. vrednotenje, analiza, izbira alternativ
5. realizacija odločitve



Slika 12. Večparametrski odločitveni model [1].

## 4.2 Ocenitev kriterijev

V tej fazi določimo ocene za posamezne lastnosti, na osnovi katerih bomo ovrednotili variante in zasnovali strukturo odločitvenega modela [15].

Ocene so merila, s pomočjo katerih ovrednotimo parametre alternative. Inačice razgradimo na posamezne attribute, katere ocenimo posamezno [1]. Z analizo podatkov tečajev iz preteklih let smo ovrednotili posamezne lastnosti alternativ.

Pri izkušnosti imamo opredeljeno zalogo vrednosti glede na število tečajev, pri katerih je bil inštruktor prisoten. Število tečajev, pri katerih je inštruktor že sodeloval, se tako deli na tri dele. Če je inštruktor sodeloval na manj kot petih tečajih, dobi oceno slabo, če je bil prisoten na vsaj petih in manj kot desetih tečajih dobi oceno srednje in če je sodeloval pri vsaj desetih tečajih, dobi oceno dobro.

Pri oddaljenosti, se ocena deli glede na zaloge vrednosti kilometrov, ki merijo oddaljenost inštruktorjevega naslova in naslova, kjer se bo odvijal tečaj. Ocena je razdeljena na tri prirejene vrednosti. Dobra, kjer je oddaljenost manjša od petnajstih kilometrov, srednja, kjer je oddaljenost višja od petnajstih kilometrov, vendar nižja od trideskih ter slaba, kjer je oddaljenost višja od trideskih kilometrov, kot prikazuje koda v nadaljevanju.

```
//ocenimo oddaljenost
$oddOcena = 0;

if ($oddaljenost < 15) {
    $ oddOcena = 3;
}
elseif (15 <= $oddaljenost && $oddaljenost < 30) {
    $ oddOcena = 2;
}
else{
    $ oddOcena = 1;
}
```

Stopnja znanja je že ocena sama po sebi, saj se deli na tri zaloge vrednosti in sicer na višjo, srednjo in nižjo. Pri znanju se upošteva njihovo poznavanje določene veščine. Torej, če je njihovo znanje za učenje temeljnih postopkov oživljanja odlično, ima prav tako tudi oceno pri tej lastnosti.

Lasten prevoz ima le dve vrednosti v zalogi vrednosti in sicer da ali ne. Če ima inštruktor lasten prevoz, dobi parameter vrednost 1, sicer pa 0.

### 4.3 Funkcija koristnosti

Funkcija koristnosti  $F$  je predpis po katerem se vrednosti posameznih parametrov  $X_i$  združujejo v spremenljivko  $Y$ , ki ponazarja končno oceno ali koristnost variante. Vsako varianto imenovano  $a_i$  opišemo z vrednostmi osnovnih kriterijev [11].

Odločitveni sistem se odloča glede na ocene atributov in ciljev pri izbiri prioritete, ki smo jih vnesli v obliki uteži pri dodajanju novega tečaja. Uporabo uteži smo uvedli, da lahko inštruktor z njimi doseže svoje prioritete pri izbiri predavateljev. Večjo prioriteto, oziroma utež damo pomembnejšim kriterijem, kateri bi morali imeti višji vpliv na končno funkcijo koristnosti. Z različno uporabo uteži lahko drastično spremeni končni rezultat odločilnega sistema. V našem primeru je cilj izbira najboljšega inštruktorja, ki ima najboljše lastnosti pri željenih prioritetah.

Koristnost izračunamo tako, da vzamemo oceno znanja, ki ga ima neka oseba za določeno skupino in ga zmnožimo z utežjo, ki jo je inštruktor določil, ko je dodajal nov tečaj. Temu prištejemo zmnožek ocene izkušenj, ki jih ima inštruktor za ta tip skupine s svojo utežjo. Dobljenemu prištejemo oceno oddaljenosti, ki jo zmnožimo skupaj z lastno utežjo in na koncu prištejemo še faktor lastnega prevoza.

Za boljšo predstavo enačba v nadaljevanju prikazuje končno funkcijo koristnosti.

$$\begin{aligned} \text{Funkcija koristnosti} = & (\text{utež znanja} \times \text{kriterij znanja}) + (\text{utež izkušenj} \times \text{kriterij izkušenj}) + \\ & + (\text{utež oddaljenosti} \times \text{kriterij oddaljenosti}) + (\text{prevoz} \times \text{utež prevoza}) \end{aligned}$$

Kot primer vzamemo enostaven tečaj, kjer bo tečajnikov malo, vendar bo zelo oddaljen od prebivališča večine tečajnikov. V tem primeru bi se zdelo smiselno, da bi pri izbiri uteži izbrali kot najpomembnejšo prioriteto čim manjšo oddaljenost. Kasneje pa sledijo še lasten prevoz, znanje in izkušnost.

#### 4.4 Rangiranje in izbira variant

Ko dobimo vse končne izračune funkcije koristnosti za alternative, lahko variante začnemo rangirati. Najprej rangiramo variante za eno skupino tečaja. Preverimo pri najboljšem, da inštruktor še ni nikamor dodeljen ter mu pošljemo vabilo. Nato se premaknemo na naslednjo skupino tečaja in to nadaljujemo dokler imamo prosta mesta. Ko zapolnimo vsa mesta in določimo vse izbrane variante, ponudimo tudi možnost zamenjave, če smo mnenja, da odločitveni sistem ni izbral najbolj optimalnih alternativ.

## Poglavje 5 Sklepne ugotovitve

V diplomski nalogi smo razvil spletno aplikacijo, ki omogoča inštruktorjem lažje izbrati inštruktorje za izvedbo tečajev prve pomoči. Cilj naloge je bil izboljšava in prenos sistema iz ročnega zapisovanja v tabele Excel v sodobno spletno aplikacijo, ki bi nudila večji nadzor in boljšo kontrolo udeležbe inštruktorjev na tečajih. Končna rešitev drastično pohitri proces izbire inštruktorjev, od začetnega pregledovanja možnih kandidatov (študentov) pa do izbire najprimernejših in nato obveščanje le teh.

Pri sami aplikaciji smo naleteli na veliko dela, še predno smo se lotil samega odločitvenega sistema, saj je bilo potrebno aplikacijo narediti prijazno uporabniku. Čim bolj enostavno, pa vseeno z zadostno mero funkcionalne uporabe. Spletna aplikacija je narejena tako, da deluje na večinoma vseh spletnih brskalnikih.

V nadaljevanju razvoja bo na vrsto prišla selitev z lokalnega na spletni strežnik. Prav tako bi si želeli odpraviti pomanjkljivosti na varnostem področju ter zmanjšati število robnih primerov. V primeru pozabljenega gesla, bi bilo potrebno implementirati tudi možnost spremembe gesla. Pri vpisu naslova lahko pride do napačnega razumevanja in sistem vzame napačen kraj, kar lahko privede do napak. Dodali bi lahko tudi kakšno funkcionalnost več. Pri posameznih tečajih bi bilo dobrodošlo objavljane komentarjev, kjer bi se lahko inštruktorji zmenili za kakšno podrobnost, prav tako bi na tej zasloni maski lahko implementiral zemljevid, ki bi kazal točno lokacijo tečaja. Poleg tega bi lahko izpopolnil samo obliko spletne strani, da bi bila uporabniku še bolj prijazna.

Pri registraciji inštruktorja bi lahko izpopolnil sistem za določanje parametra znanje. Inštruktorji na začetku izberejo kakšno stopnjo znanja posedujejo, tukaj pa bi lahko prišlo do zlorabe, oziroma negotovosti v kateri kriterij spadajo. Dodan bi moral biti kratek test preverjanja znanja za lažjo opredelitev.

Pri odločanju se pojavi tudi problem v primeru premajhnega števila inštruktorjev, kajti en inštruktor lahko naenkrat poučuje le v eni skupini tečaja in se težje določi optimalno izbiro.



## Literatura

- [1] Bohanec, M.: Odločanje in modeli, DMFA – založništvo (2006), ISBN 961-212-190-7.
- [2] Crockford, D.: JavaScript: The Good Parts, O'Reilly Media (2008), ISBN 978-0-596-51774-8.
- [3] Lockhart, J.: Modern PHP: New Features and Good Practices, O'Reilly Media (2015), ISBN 978-1-4919-0501-2.
- [4] Nixon, R.: Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5, O'Reilly Media (2014), ISBN:978-1-4919-1866-1.
- [5] Turban, E.: Decision support and expert systems, Macmillian (1988), ISBN-13 978-0024216502.

### Spletni viri:

- [6] Virginia Tech, Decision Support System, Dostopno 25.1.2016 na:  
[http://www.bit.vt.edu/academics/dss\\_index.html](http://www.bit.vt.edu/academics/dss_index.html).
- [7] jQuery – write less, do more, Dostopno 12.12.2015 na:  
<https://jquery.com/>.
- [8] Google Maps Javascript API, Distance Matrix Service, Dostopno 15.10.2015 na:  
<https://developers.google.com/maps/documentation/javascript/distancematrix>.
- [9] MySQL Documentation, Dostopno 14.9.2015 na:  
<http://dev.mysql.com/doc/>.
- [10] Wikipedia, Sublime Text, Dostopno 10.8.2015 na:  
Dosegljivo [10.8.2015]: [https://en.wikipedia.org/wiki/Sublime\\_Text](https://en.wikipedia.org/wiki/Sublime_Text).

- [11] LAMP (Linux, Apache, MySQL, PHP), Dostopno 10.8.2015 na:  
<http://searchenterprise-linux.techtarget.com/definition/LAMP>.
- [12] Apache Friends, XAMPP, Dostopno 10.8.2015 na:  
<https://www.apachefriends.org/index.html>.
- [13] phpMyAdmin, Dostopno 20.1.2016 na:  
<https://www.phpmyadmin.net/>.
- [14] Večparametrski odločitveni sistemi, Dostopno 3.1.2016 na:  
<http://kt.ijs.si/MarkoBohanec/org95/Uvod.html>
- [15] Faze odločitvenega procesa, Dostopno 3.1.2016 na:  
<http://kt.ijs.si/MarkoBohanec/org95/Faze.html>